Assessing Student Response in Tutorial Dialogue Context using Probabilistic Soft Logic

Rajendra Banjade
The University of Memphis
Memphis, TN, USA
rajbanjade@gmail.com

Vasile Rus
The University of Memphis
Memphis, TN, USA
vrus@memphis.edu

ABSTRACT

Automatic answer assessment systems typically apply semantic similarity methods where student responses are compared with some reference answers in order to access their correctness. But student responses in dialogue based tutoring systems are often grammatically and semantically incomplete and additional information (e.g., dialogue history) is needed to better assess their correctness. In that, we have proposed augmenting semantic similarity based models with, for example, knowledge level of the student and question difficulty and jointly modeled their complex interactions using Probabilistic Soft Logic (PSL). The results of the proposed PSL models to infer the correctness of the given answer on DT-Grade dataset show the more than 7% improvement on accuracy over the results obtained using a semantic similarity model.

Keywords

Tutoring System, Answer Assessment, Probabilistic Soft Logic

1. INTRODUCTION

Open ended answers are responses produced by students to questions, e.g. in a test or in the middle of a tutorial dialogue. Such answers are very different from answers to multiple choice questions where students just choose one or more options from the given choices and they are more easier to evaluate than open ended answers. In conversational Intelligent Tutoring Systems (ITSs; [18, 14]), the systems should be able to assess the students' responses in order to provide them appropriate feedback and to plan the subsequent part of the dialogue.

The true understanding of student answers is intractable as it requires collecting and doing reasoning over a huge knowledge, including the linguistic knowledge, domain knowledge, and world knowledge. As a practical alternative, semantic similarity methods are applied [5, 10, 15]. In this approach, systems assess student responses by measuring how much of the targeted concept is present in the student answer. Accordingly, the subject matter experts create target (or reference) answers to the questions that students will be prompted to answer and the system assesses how much of the targeted concept is present in the student answer by measuring the semantic similarity between student's answer with reference answer.

The meaning of the reference answer is known because they are created by subject matter experts. The high similarity between student answer with reference answer indicates that the answer is correct. Otherwise, the answer is partially correct, or incorrect. This approach has been widely used in understanding student responses in tutoring systems and in automatic answer assessment systems in general (see Section 2). It is fast, does not require too much of information, and has been often found to be effective.

However, the implied assumption in similarity based answer assessment approach is that the student answer and the reference answer are self contained (i.e., grammatically and semantically complete). But student responses in conversational tutoring systems vary a lot as illustrated in Table 1. The meanings of students' responses often depend on the dialogue context and problem/task description. For example, students frequently use pronouns, such as *they, he, she*, and *it*, in their response to tutor's questions or other prompts. In an analysis of tutorial conversation logs, Niraula *et al.*[13] found that 68% of the pronouns used by students were referring to entities in the previous utterances or in the problem description. In addition to anaphora, complex coreferences are also employed by students.

Furthermore, in tutorial dialogues students react often with very short answers which are easily interpreted by human tutors as the dialogue context offers support to fill-in the blanks or untold parts. Such elliptical utterances are common in conversations even when the speakers are instructed to produce more syntactically and semantically complete utterances [4]. By analyzing 790 student responses given to DeepTutor tutoring system [14], we have found that about 25% of the times even human needed additional information, such as the dialogue history in order to properly assess them [2].

As illustrated in Table 1, the student answers may vary greatly. For instance, answer $A1$ is elliptical. The *bug* in $A2$ is referring to the mosquito and *they* in $A3$ is referring

Rajendra Banjade and Vasile Rus "Assessing Student Response in Tutorial Dialogue Context using Probabilistic Soft Logic" In: *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019)*, Collin F. Lynch, Agathe Merceron, Michel Desmarais, & Roger Nkambou (eds.) 2019, pp. 258 - 263

Table 1: Some student answers to the given question asked at some point during interactions with DeepTutor.

Problem description: A car windshield collides with a mosquito, squashing it.
Tutor question: How do the amounts of the force exerted on the windshield by the mosquito and the force exerted on the mosquito by the windshield compare?
Reference answer: The force exerted by the windshield on the mosquito and the force exerted by the mosquito on the windshield are equal and opposite.
Student answers:
A1. *Equal*
A2. *The force of the bug hitting the window is much less than the force that the window exerts on the bug*
A3. *they are equal and opposite in direction*
A4. *equal and opposite*

to the amount of forces exerted to each other. Due to such variations in the answers, the semantic similarity methods alone can have issues in properly assessing those answers. For instance, the similarity between answer $A1$ and the reference answer will be very low.

In this paper, we present Probabilistic Soft Logic (PSL; [3]) model for improving automatic assessment of open-ended answers in conversational ITS by augmenting the semantic similarity model with additional information, such as question difficulty and the knowledge level of the student. For instance, a high knowledge student answering many of the difficult questions correctly will probably answer the current question correctly. The proposed method allows us to model the complex interactions between the stochastic variables, such as student's knowledge level, question difficulty and the correctness of the student answer. In specific, the proposed PSL model which works on probabilistic reasoning framework allows us to concisely express our knowledge in First Order Predicate Logic (FOPL) rules and to provide the extent of our belief on such knowledge as weights. The inference is done over Probabilistic Graphical Model (PGM).

We evaluated our models on a dataset consisting of 790 responses collected during DeepTutor experiments and annotated for their correctness. The results show that augmenting the similarity model with question difficulty and knowledge level of the student improved the accuracy of our answer assessment model by about 8% when compared to results obtained using only the semantic similarity information.

2. RELATED WORK

Our work is more focused on assessing student responses in conversational tutoring systems. But most of the existing work has been performed on standard test taking environment (e.g., assignment checking). In this section, we briefly discuss approaches for constructed answer assessment where the student answers are short (one to just few lines) and the reference answers are available to compare with.

Martin *et al.* [9] proposed an assessment system OLAE using Bayesian nets. Latent Semantic Analysis (LSA; [8]) and machine translation evaluation methods are also applied for answer grading. LSA method was also used in AutoTutor system [7].

Various researches show that the similarity based methods can be potentially used in the answer grading tasks [10, 15, 11]. In fact, a Semantic Evaluation (SemEval) shared task called Joint Student Response Analysis and 8th Recognizing Textual Entailment Challenge was organized in 2013 [5] to promote and streamline research in this area and almost all of the participating teams applied semantic similarity and textual entailment techniques.

Although various results show that the similarity based methods can be used in answer grading tasks, their implied assumption is that the text available are standard texts with noise filtered. Our work is focused on using naturally occurring texts from conversational tutoring systems where various linguistic phenomena are present, such as coreferences and ellipsis as discussed in Section 1. We also augment the semantic similarity based model using additional knowledge.

Furthermore, various datasets have been published over the years [12, 10, 5, 17]. But dataset from conversational systems with additional information (e.g., previous utterance, problem description, knowledge level of the student, question difficulty) are very limited. We annotated 790 student responses collected during an experiment with DeepTutor [14]. The dataset is made available for research purpose [2].

3. DATASET

We created the DT-Grade dataset [2] by extracting student answers from logged tutorial interactions between 36 junior level college students and the DeepTutor system [14]. During the interactions, each student solved 9 conceptual physics problems and the interactions were in the form of purely natural language dialogues, i.e., with no mathematical expressions and special symbols. We selected 790 answers for the annotation. We chose the more difficult ones (by observing responses from some students, the nature of the question, and so on) such that the similarity based models alone will have difficulty judging those answers.

Table 2: Summary of DT-Grade dataset.

Label	Count
All	790
Correct	319 (40.379%)
Correct but incomplete	292 (36.962%)
Incorrect	179 (22.658%)

Each instance contains the following information: (a) problem description (describes the scenario or context), (b) tutor question, (c) student answer in its natural form (i.e., without correcting spelling errors and grammatical errors), (d) list of reference answers for the question and has been assigned one of the following labels.

1. **Correct**: Answer is fully correct in the context. Extra information, if any, in the answer is not contradicting with the answer.
2. **Correct-but-incomplete**: Whatever the student provided is correct but something is missing, i.e. it is not complete. If the answer contains some incorrect part also, the answer is treated as incorrect.
3. **Incorrect**: Student answer is incorrect.

The dataset and further details about the collection and annotation of it can be found at [2].

4. PROBABILISTIC SOFT LOGIC MODELS

4.1 Background

Probabilistic Soft Logic (PSL; [3]) is an approach to combining knowledge in the form of first-order logic rules and probability in a single representation. It forms Probabilistic Graphical Models (PGM) which allow us to efficiently handle uncertainty and first-order logic allows us to compactly represent the knowledge. Furthermore, it allows us to jointly model the complex interactions among stochastic variables. For example, voting decision of friends has some influence on each other. Similarly, in answer assessment, a high knowledge student giving correct answers to the difficult questions will probably answer another difficult or easy question correctly and we can model such knowledge in a PSL model. On the other hand, typical machine learning algorithms assume that the data bear *i.i.d.* properties.

First-Order Knowledge Base. A first-order knowledge base (KB) is a set of formulas in first order logic [6]. Formulas are constructed using symbols: *constants*, *variables*, *predicates*, and *functions*. Constant represents an object (e.g., John). Functions represent mappings from tuples of objects to objects (e.g., $FatherOf$). Predicate represents relations among objects (e.g., $Friends$) or attributes of objects (e.g., $Smokes$). The formulas are typically written in clausal form (also known as conjunctive normal form (CNF)). For example,

$$Friends(x, y) \wedge Friends(y, z) \rightarrow Friends(x, z)$$

PSL Program. A PSL program consists of rules along with relative weights associated with them and the data (or observations). The weights in the following example rules are assigned quite arbitrarily but they can be learned from the data which we discuss later.

$$5.0 : Friends(x, y) \wedge Friends(y, z) \rightarrow Friends(x, z)$$
$$2.0 : Friends(x, y) \wedge Colleague(y, z) \rightarrow Friends(x, z)$$

The rules are grounded using observations, i.e., each variable in the rules is assigned to all possible values in the observed data. For example, if there are three people: Joe, Bob, and Lili, then a grounded rule would look like,

$$5.0 : Friends(Joe, Bob) \wedge Friends(Bob, Lili) \rightarrow Friends(Joe, Lili)$$

Predicates in PSL program can have truth values in the range of [0 1], i.e. they are soft. For example, if it is not sure about the friendship of Joe and Bob but there is some possibility, then this uncertainty can be defined as a truth value in the range of 0 to 1. This is different from Markov Logic Network (MLN) where the predicates can have truth values either true or false (i.e., constraints in MLN are harder than PSL).

Prior Knowledge. The prior knowledge can also be encoded as rules in the PSL program. In our hypothetical example, let's assume that people who are neither friends of friends nor friends of colleagues can still be friends but the chances are very low. This can be expressed in the PSL program as illustrated below. It should be noted that the weight to our prior is very low as our belief is that any two persons being friends to each other (given no additional information) is possible but very less likely.

$$0.0001 : Friends(x, z)$$

As mentioned, the weights to the rules can be learned from the data itself. We discuss on this later. Next, we discuss some of the variables, predicates and rules we used in our PSL program (or model).

4.2 Model

Variables. Our model has two variables (s and a) and by convention, the variables are represented by lower case letters.

s - Student id, a - Answer id (or just id) which uniquely identifies an instance in the dataset. It should be noted that the question belonging to a may be same as that of some other answer id b because the same set of problems were attempted by multiple students.

Predicates. Following are the predicates used in our model.

- $SimilarityHigh(a) \in \{0, 1\}$ - similarity of answer a with corresponding reference answer is high
- $SimilarityMedium(a) \in \{0, 1\}$ -similarity of answer a with corresponding reference answer is medium

- $SimilarityLow(a) \in \{0, 1\}$ - similarity of answer a with corresponding reference answer is low
- $PriorKHigh(s) \in \{0, 1\}$ - prior knowledge of the student s is high
- $PriorKMedium(s) \in \{0, 1\}$ - prior knowledge of the student s is medium
- $PriorKLow(s) \in \{0, 1\}$ - prior knowledge of the student s is low
- $QDifficultyHigh(a) \in [0\ 1]$ - question difficulty is high (fraction of students who answered the question corresponding to a incorrectly)
- $QDifficultyMedium(a) \in [0\ 1]$ - question difficulty is medium (fraction of students who answered the question corresponding to a correctly but incompletely)
- $QDifficultyLow(a) \in [0\ 1]$ - question difficulty is low (fraction of students who answered the question corresponding to a correctly)
- $AttemptedBySameStudent(a, b) \in \{0, 1\}$ - whether a and b were attempted by the same student
- $Correct(a) \in [0\ 1]$ - the truth value of answer a being correct
- $CorrectButIncomplete(a) \in [0\ 1]$ - the truth value of answer a being correct but incomplete
- $Incorrect(a) \in [0\ 1]$ - the truth value of answer a being incorrect

We have created three predicates for semantic similarity and for prior knowledge to avoid biases in assigning weights in some rules. For example, if we have rules $3.0 : Similarity(a) \rightarrow Correct(a)$ and $2.0 : Similarity(a) \rightarrow Incorrect(a)$, then low value of similarity score will still favor the first rule.

Rules and Priors. We present few rules with quite arbitrary weights. We learn the weights for those rules from the data which we present later in this section. The priors (starting with negation symbol $\sim$) specify the possibilities of being false. It should be noted that the weights are relative to each other and do not have to sum up to 1.

$2.0 : SimilarityHigh(a) \wedge QdifficultyLow(a) \rightarrow Correct(a)$

$3.0 : SimilarityLow(a) \rightarrow Incorrect(a)$

...

$0.002 :\sim Correct(a)$

$0.004 :\sim CorrectButIncomplete(a)$

$0.003 :\sim Incorrect(a)$

4.3 Data

We used DT-Grade dataset described in Table 2. It includes responses provided by 36 students and we also had pretest scores for them. The pretest was a multiple-choice test which consisted of 39 questions.

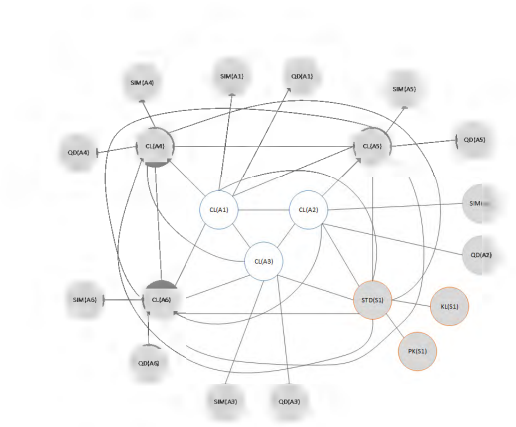

Figure 1: An illustration of a grounded probabilistic graphical network for a student. The shaded nodes are *evidence* nodes and non-shaded nodes are *query* nodes. CL - Correctness label, QD - Question difficulty, STD - Student, KL - Knowledge level, SIM - Similarity.

4.4 Grounding

During grounding phase, all the variables in the rules are substituted with possible values from the observations (i.e., data). Figure 1 illustrates an example of a grounded graphical network for a student's data but the graph can grow very large. For instance, the nodes corresponding to correctness labels of each answer are actually 3 (*Correct, CorrectButIncomplete*, and *Incorrect*) but in the graph they are represented by a single node CL. Similarly, the question difficulty QD has three values (high, medium, and low) and each one is actually represented by a separate node. Also, each student has attempted around 20 questions in average (counting those in the DT-Grade dataset only) which makes the graph bigger than what is shown in the figure. We discuss on the scale of the network in Weight Learning section.

The shaded nodes in the graph are observed nodes which we call **evidence**, whereas the light nodes are **query** nodes. During inference, the truth values of the query nodes are predicted jointly based on the evidence.

The similarity between student answer and the corresponding reference answer was calculated using optimal word alignment based method which has performed very well in general. We used the methods implemented in SEMILAR library [16]. We then grouped the similarity scores into high (score $>$ 0.5), medium (0.5 $\geq$ score $>$ 0.35), and low ($\leq$ 0.35) using empirically chosen threshold values. Similarly, we grouped the prior knowledge of the students into high ($>$ 0.8), medium (0.8 $\geq$ score $>$ 0.5), and low ($\leq$ 0.5) based on their pretest scores.

We calculated the question difficulty (high, medium, and low) as discussed in Section 4.2 (predicate definitions related to question difficulty). However, for question difficulty we have used soft values. In specific, each question has soft value (in [0 1]) for each of the difficulty levels: high, medium, and low. But for the difficult question, for example, the truth value of the predicate *QDifficultyHigh(a)* will have higher value than the truth value of the predicates corresponding to other difficulty levels (medium, and low).

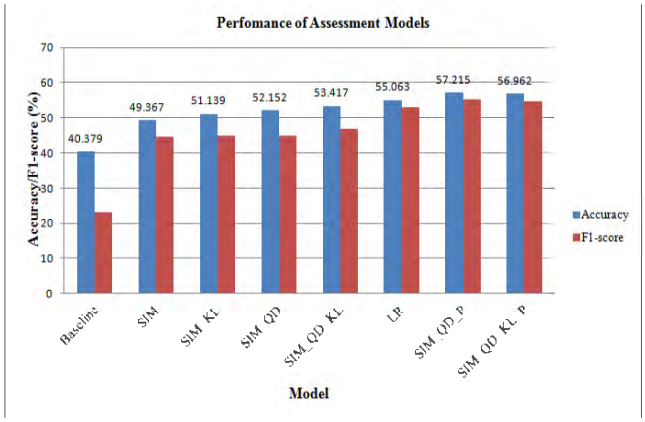

Figure 2: Results of different Probabilistic Soft Logic models on DT-Grade dataset. The models ending with _P used the prior information learned using Logistic Regression (LR) models.

4.5 Weight Learning for PSL Rules

Laving out the internal details, the PSL rules' weight learning process is similar to typical supervised model learning process. We provide the ground truth (human annotated correctness label of the given answer) for the query predicate (which corresponds to a node in the grounded network graph). For each answer, there will be three query predicates one for each of the three labels. Since the labels are mutually exclusive, only one of them is set to 1.0 while other two will be set to 0.0. As we discussed earlier, the grounded network can become very large depending on the set of rules and the size of the dataset used to ground the rules.

Also, the same node cannot be a query node as well as evidence node at the same time. Therefore, we have replicated each student's data several times and renamed their ids such that we can make each answer in the original set (though renamed) a query at one time while using it as an evidence when other nodes are query nodes. This is important for us in both training (i.e., learning rules' weights) and evaluation phase because the dataset we used is comparatively small. For instance, if we make one answer for each student a query and keep others as evidence, then we will have only 36 records for the weight learning as well as for the evaluation. But making each node a query node at least once, we have the full dataset which is several times bigger than the aforementioned size and we can also evaluate our model using the full dataset (for example, by using leave-one-out approach). In another words, this process allows us to utilize the full set of data.

Just to get a sense of the scale of the graph, we assume that each student's data is replicated 5 times. Then the size of the graph (by taking the dominant term only) will be $(5 * 790) * (5 * 790) \sim 15\ million$. Weight learning in such a huge probabilistic graphical model is impossible at least in our experimental settings. Therefore, we have pruned some rules that rapidly increase the size of the graph (e.g. the rules of the type: if answer to a is correct, then answer to b is also correct for the given student) and the resulting graph had about 200,000 nodes, on which we have managed to learn the weights for the rules.

For those rules which rapidly increase the size of the network with increasing size of the data, we have learned the weights for each student and estimated the weights of the rules using weights learned at student level which is the sub-optimal solution. For each student, the average size graph had only few thousand nodes.

5. RESULTS

Including semantic similarity and additional information, we built several PSL models. For the experiments, we used the PSL tool[1] developed at University of Maryland, College Park. The tool uses Hinge-Loss Markov Random Fields (HL-MRFs) for inference and weight learning [1]. We set the number of iterations to be performed by the optimizer to 50,000.

By assuming that the performance of a student is independent of others, we refactored the graph into subgraphs one for each student and took the leave-one-student-out approach for PSL rules' weight learning and evaluation. As discussed in Section 4.5, we learned the weights for the rules from 35 students at a time (except for few rules for which weights were estimated using weights learned student-wise) and applied to the leave out student. Performing inference in such smaller graphs is computationally very efficient (takes few seconds for each student when run in a normal workstation). Also, the question difficulty was calculated based on training data only, i.e. using 35 students' data at a time.

Once inference is complete, i.e. the truth values for $Correct$, $CorrectButIncomplete$, and $Incorrect$ predicates are assigned for each query answer. We then chose the correctness label corresponding to the highest truth value among those three. It should be noted that the truth value for each of them was in the interval [0 1] but their sum does not have to be 1.0. Then, we calculated the accuracy and F1 scores. The results of our various models are presented in Figure 2.

The baseline system is the majority class classifier, i.e. which labels each answer as correct. The accuracy of this baseline model was 40.379% which is equivalent to the percentage of correct answers in the dataset. SIM model used the similarity information only. It obtained 9% improvement over the baseline. As mentioned earlier, the DT-Grade dataset was developed by selecting the difficult cases, particularly difficult to judge by only comparing the student answer with the reference answer. Therefore, we consider 9% improvement in accuracy over baseline results as a notable improvement.

We then augmented the model using knowledge level (KL) of the student and question difficulty (QD). The KL includes the prior knowledge of the student which was assessed using multiple choice questions. The results were improved after adding question difficulty and knowledge level separately. Furthermore, when combined together our model achieved 53.417% accuracy which is above 4% improvement over results obtained using similarity information only.

In an another experiment, we used the priors learned using

[1]http://psl.linqs.org/

Logistic Regression (LR). In specific, we obtained the probabilities (precisions) of any answer being *Correct*, *CorrectButIncomplete*, and *Incorrect* based on correctness predictions made by LR model when the given set of rules were used as features and used them in our PSL models as priors (model names ending *_P*). Since the priors in PSL models needed to be in negated form, we deducted each probabilities learned from LR from 1.0 and used the resulting values as priors. This has improved the results by about 5% in *SIM_QD* model and about 3% in *SIM_QD_KL* model. The results are above 7% when compared to *SIM* model. These results are also better when compared to the results of LR model itself. This shows that the LR model which is very different from PSL can complement the PSL model.

We learned priors using LR model only for *SIM_QD*. The *SIM_KL* included pretest scores as well as rules of the type: if answer to a is correct, then answer to b is also correct for the given student. Such rules that capture the relational dependencies are not easily modeled in Logistic Regression. Furthermore, the results of *SIM_QD_KL_P* is slightly less than *SIM_QD_P*. It seems that the concordance between the weights of the PSL rules and the priors learned separately may not be perfect in some cases.

6. CONCLUSION

We presented joint learning models using Probabilistic Soft Logic (PSL) for improving the assessment of open-ended student responses in conversational tutoring systems where the student responses can vary a lot. Specifically, our models augmented semantic similarity information with non-linguistic knowledge (student's knowledge level and question difficulty) and improved the accuracy of the assessment model when evaluated with DT-Grade dataset. The accuracy of our model using informed priors was up to 57.215%, which is more than 7% improvement over the results of semantic similarity based models. In the future, we intend to add additional information in the model and improve on PSL rules' weight learning by clustering students' data.

7. ACKNOWLEDGMENTS

This work was partially supported by The University of Memphis, the National Science Foundation (awards CISE-IIS-1822816 and CISE-ACI-1443068), and a contract from the Advanced Distributed Learning Initiative of the United States Department of Defense.

8. REFERENCES

[1] S. H. Bach, M. Broecheler, B. Huang, and L. Getoor. Hinge-loss markov random fields and probabilistic soft logic. *arXiv preprint arXiv:1505.04406*, 2015.

[2] R. Banjade, N. Maharjan, N. B. Niraula, D. Gautam, B. Samei, and V. Rus. Evaluation dataset (dt-grade) and word weighting approach towards constructed short answers assessment in tutorial dialogue context. 2016.

[3] M. Brocheler, L. Mihalkova, and L. Getoor. Probabilistic similarity logic. *arXiv preprint arXiv:1203.3469*, 2012.

[4] J. G. Carbonell. Discourse pragmatics and ellipsis resolution in task-oriented natural language interfaces. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics*, pages 164–168. Association for Computational Linguistics, 1983.

[5] M. O. Dzikovska, R. D. Nielsen, C. Brew, C. Leacock, D. Giampiccolo, L. Bentivogli, P. Clark, I. Dagan, and H. T. Dang. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Technical report, DTIC Document, 2013.

[6] M. R. Genesereth and N. J. Nilsson. Logical foundations of artificial. *Intelligence. Morgan Kaufmann*, 58, 1987.

[7] A. C. Graesser, P. Wiemer-Hastings, K. Wiemer-Hastings, D. Harter, T. R. G. Tutoring Research Group, and N. Person. Using latent semantic analysis to evaluate the contributions of students in autotutor. *Interactive learning environments*, 8(2):129–147, 2000.

[8] T. K. Landauer. Automatic essay assessment. *Assessment in education: Principles, policy & practice*, 10(3):295–308, 2003.

[9] J. Martin and K. VanLehn. Student assessment using bayesian nets. *International Journal of Human-Computer Studies*, 42(6):575–591, 1995.

[10] M. Mohler and R. Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 567–575. Association for Computational Linguistics, 2009.

[11] N. Murrugarra, S. Lu, and M. Li. Automatic grading student answers. 2013.

[12] R. D. Nielsen, W. Ward, J. H. Martin, and M. Palmer. Annotating students' understanding of science concepts. In *LREC*, 2008.

[13] N. B. Niraula, V. Rus, R. Banjade, D. Stefanescu, W. Baggett, and B. Morgan. The dare corpus: A resource for anaphora resolution in dialogue based intelligent tutoring systems. In *LREC*, pages 3199–3203, 2014.

[14] V. Rus, S. D'Mello, X. Hu, and A. Graesser. Recent advances in conversational intelligent tutoring systems. *AI magazine*, 34(3):42–54, 2013.

[15] V. Rus and M. Lintean. A comparison of greedy and optimal assessment of natural language student input using word-to-word similarity metrics. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 157–162. Association for Computational Linguistics, 2012.

[16] V. Rus, M. C. Lintean, R. Banjade, N. B. Niraula, and D. Stefanescu. Semilar: The semantic similarity toolkit. In *ACL (Conference System Demonstrations)*, pages 163–168. Association for Computational Linguistics, 2013.

[17] J. Z. Sukkarieh and E. Bolge. Building a textual entailment suite for the evaluation of automatic content scoring technologies. In *LREC*. Citeseer, 2010.

[18] K. VanLehn, A. C. Graesser, G. T. Jackson, P. Jordan, A. Olney, and C. P. Rosé. When are tutorial dialogues more effective than reading? *Cognitive science*, 31(1):3–62, 2007.